

**UNIVERSIDAD AUTÓNOMA DE MADRID**  
**ESCUELA POLITÉCNICA SUPERIOR**



**Grado en Ingeniería de Tecnologías y Servicios de  
Telecomunicación**

**TRABAJO FIN DE GRADO**

**APLICACIÓN ANDROID DE ESTADÍSTICAS  
DEPORTIVAS**

**Álvaro Díaz Arroyo**  
**Tutor: Antonio Rovira Moreno**  
**Ponente : Eduardo Boemo Scalvinoni**

**Julio 2017**



# **APLICACIÓN ANDROID DE ESTADÍSTICAS DEPORTIVAS**

**AUTOR: Álvaro Díaz Arroyo**  
**TUTOR: Antonio Rovira Moreno**

**Digital System Laboratory**  
**Dpto. Tecnología Electrónica y de Comunicaciones**  
**Escuela Politécnica Superior**  
**Universidad Autónoma de Madrid**  
**Julio de 2017**



# Resumen

Este Trabajo de Fin de Grado consiste en el diseño y desarrollo de una aplicación para dispositivos móviles con Sistema Operativo Android. Se basa en el análisis y tratamiento masivo de datos relativos a eventos deportivos de fútbol y baloncesto, para la obtención de estadísticas orientadas a utilizarse en relación con los servicios que ofrecen las casas de apuestas deportivas.

La aplicación recopila datos tales como:

- Goles
- Saques de esquina
- Tarjetas
- Clasificaciones históricas
- Calendario de encuentros
- Información de partidos
- Información de equipos

Lo expuesto se particulariza a equipos, partidos (dos equipos), temporadas y ligas. Los datos han sido extraídos de diversas fuentes y en la actualidad suman 5000 de partidos y 400 de equipos.

Además, la aplicación cuenta con las siguientes funcionalidades:

- Clasificaciones de fútbol de las ligas más importantes a nivel europeo
- Clasificaciones de baloncesto de las ligas más importantes a nivel mundial
- Estadísticas detalladas de cada equipo en ambas disciplinas
- Calendario para mostrar los partidos que hay en el día
- Función para filtrar por días el calendario

La aplicación recibe el nombre de Estadísticas deportivas y estará disponible para su descarga totalmente gratuita en la plataforma de aplicaciones Google Play Store.



# Abstract

This Bachelor Thesis consists on the design and developing of an app for Android mobile devices. The app is based on the analysis and massive treatment of data related to football and basketball events to obtain certain statistics. These statistics are thought to maximise their usage and usefulness regard to the markets offered by the different bookmakers.

The app collects data such as:

- Goals
- Corners
- Cards
- Historical Standings
- Match Schedule
- Match information
- Team information

The above is particularized to teams, matches (two teams), seasons and leagues. The data have been extracted from various sources and currently Total 5000 matches and 400 teams.

In addition, the application has the following functionalities:

- Football Standings of the most important European leagues
- Basketball Standings of the most important Leagues Worldwide
- Detailed statistics of each team in both disciplines
- Calendar to show the matches in the day
- Day-filter function on the calendar

This app is called “Estadísticas deportivas” and will be available for free downloading on Google Play Store.





## **Palabras clave**

Móvil, Android, deportes, fútbol, baloncesto, resultados, estadísticas, apuestas, picks.

## **Keywords**

Mobile, Android, sports, football, soccer, basketball, score, stats, betting, picks.



## ***Agradecimientos***

*En primer lugar me gustaría agradecer tanto a mi tutor Antonio Rovira, como a mi ponente Eduardo Boemo la posibilidad de realidad este proyecto.*

*A mis padres y a mi hermano por apoyarme en todo momento a lo largo de toda mi vida y toda la paciencia que han tenido en el transcurso de la carrera. Además también agradecer especialmente a Esther por aguantarme y darme soluciones para calmar todo en los días que el agobio podía conmigo.*



# ÍNDICE DE CONTENIDOS

1	Introducción .....	1
1.1	Motivación .....	1
1.2	Objetivos .....	1
1.3	Organización de la memoria .....	2
2	Estado del arte .....	3
2.1	Aplicaciones similares .....	3
2.1.1	SofaScore .....	3
2.1.2	Predicción de fútbol .....	4
2.1.3	Predicción de fútbol Forebet .....	4
3	Diseño .....	7
3.1	Objetivos de diseño .....	7
3.2	Requisitos de diseño .....	7
3.2.1	Sistemas operativos compatibles .....	8
3.2.2	Versiones compatibles .....	10
3.2.3	Pantallas compatibles .....	10
3.3	Diagrama de bloques del diseño .....	13
3.4	Estructura de la aplicación .....	14
3.4.1	Servidor remoto .....	15
3.4.2	Splash Screen .....	15
3.4.3	Actividad principal .....	15
3.4.4	Elección de competición .....	15
3.4.5	Menú de competición .....	16
4	Desarrollo .....	17
4.1	Pasos previos .....	17
4.2	Herramientas utilizadas .....	17
4.3	Elementos principales de una aplicación .....	18
4.3.1	Activity .....	18
4.3.2	Views .....	20
4.3.3	Manifest .....	20
4.4	Elementos principales de las bases de datos .....	20
4.4.1	Base de datos .....	20
4.4.2	Tablas .....	21
4.5	Menús de la aplicación .....	21
4.6	Splash Screen .....	22
4.7	Desarrollo base de datos .....	22
4.8	Desarrollo de menú de elección de ligas .....	23
4.9	Botón retroceder .....	24
4.10	Desarrollo Clasificación .....	24
4.11	Desarrollo actividad calendario .....	26
4.11.1	Mostrar partidos .....	26
4.11.2	Calendario Partidos .....	28
5	Integración, pruebas y resultados .....	31
5.1	Añadir aplicación al Play Store .....	31
5.1.1	Generación de APK .....	31
5.1.2	Cuenta de desarrollador .....	31
5.2	Subir la aplicación a la plataforma .....	32
5.3	Plan de pruebas .....	32
6	Conclusiones y trabajo futuro .....	35

6.1 Conclusiones .....	35
6.2 Trabajo futuro .....	36
Referencias.....	I
Glosario.....	- 1 -

# ÍNDICE DE FIGURAS

FIGURA 1-1: LOGO SOFAScore .....	3
FIGURA 1-2: LOGO PREDICCIÓN DE FÚTBOL .....	4
FIGURA 1-3: LOGO PREDICCIÓN DE FÚTBOL FOREBET .....	4
FIGURA 2-1: DISTRIBUCIÓN DE DISPOSITIVOS POR SO EN 2017 .....	9
FIGURA 2-2: DISTRIBUCIÓN MÓVILES ANDROID POR VERSIÓN DE SO EN MARZO DE 2017 .....	10
FIGURA 2-3: RANGOS DE ASIGNACIÓN DE TAMAÑOS Y DENSIDADES EN FUNCIÓN DE TAMAÑOS Y DENSIDADES REALES .....	11
FIGURA 2-4: DISTRIBUCIÓN DE LOS MODELOS DE SMARTPHONE SACADOS EN 2016 POR EL TAMAÑO DE LA PANTALLA .....	12
FIGURA 2-5: DISTRIBUCIÓN DE LOS SMARTPHONES EN 2016 POR DENSIDAD DE PANTALLA .....	13
FIGURA 2-6: DIAGRAMA DE LA APLICACIÓN .....	14
FIGURA 2-7: MENÚ DE ELECCIÓN DE COMPETICIÓN .....	15
FIGURA 2-8: MENÚ DE COMPETICIÓN .....	16
FIGURA 3-1: CICLO DE VIDA DE UNA ACTIVIDAD .....	19
FIGURA 3-2: EJEMPLO DE USO RELATIVE LAYOUT .....	27
FIGURA 3-3: CÓDIGO PHP PARA LA CONSULTA DE PARTIDOS .....	27
FIGURA 3-4: CÓDIGO XML DE LA ACTIVIDAD CALENDARIO .....	28





# 1 Introducción

---

## 1.1 Motivación

Este proyecto está particularmente orientado a usuarios que realizan tanto apuestas de forma esporádica como profesional, siendo además de utilidad para todos aquellos a los que les guste el deporte, especialmente el fútbol y el baloncesto. Estos dos deportes son los más practicados y seguidos en España, lo que nos permite pensar una amplia cantidad de potenciales usuarios.

Se pretende incidir en aspectos específicos ya existentes en otras aplicaciones, con el fin de mejorarlos, siendo el principal objetivo presentar la información de los eventos deportivos más importantes a nivel mediático en España de una forma fácil, intuitiva y detallada, haciendo uso de la sencillez, comodidad y disponibilidad que nos proporcionan los dispositivos móviles, aportando:

- Base de datos histórica con más de 2000 partidos, para conseguir estadísticas valiosas.
- Poder observar de un vistazo todos los resultados de una liga.
- Estadísticas muy detalladas fáciles de entender.

## 1.2 Objetivos

La aplicación se ha pensado para recoger y mostrar estadísticas de las ligas y torneos europeos de fútbol más importantes. Y lo mismo con el baloncesto. Podrá correr sobre dispositivos que tengan sistema operativo Android con una versión superior a 4.0 (ICE\_CREAM\_SANDWICH).

El principal objetivo de este TFG es mostrar una aplicación que ofrezca al usuario la mayor cantidad y calidad de estadísticas de equipos, tanto de un partido en concreto como de una liga en general. Todas las estadísticas recopiladas de cada partido se encuentran en una base de datos alojada en un servidor que actúa de forma remota para que los dispositivos puedan tener acceso a la información de la misma todas las veces que lo deseen.

La aplicación va dirigida a toda persona que le guste la estadística deportiva, pero en concreto ha sido creada para aquellos que realizan apuestas, tanto de forma lúdica, como de forma profesional.

Ha sido especialmente diseñada para este último grupo debido a que las estadísticas que se ofrecen en la aplicación están estudiadas para maximizar las probabilidades de ganar al realizar una apuesta sea el mercado que sea, puesto que se ofrecen todo tipo de estadísticas con las que se pueden estudiar todos los tipos de apuestas que cualquier casa de apuestas puede ofertar.

La aplicación será subida a la plataforma de Google Play Store y estará disponible para su descarga totalmente gratuita.

Desde un punto de vista educativo, los objetivos principales son el aprendizaje y dominio de diferentes herramientas de programación como Android, SQL, PHP y otros lenguajes.

### **1.3 Organización de la memoria**

Debido a la multitud de términos técnicos en inglés se ha optado por no utilizar cursiva al utilizarlos.

Este documento tiene como objetivo exponer el desarrollo del Trabajo de Fin de Grado sobre una aplicación para el sistema operativo Android, con el fin de mostrar estadísticas deportivas en el fútbol y baloncesto. Para este fin se sigue la siguiente estructura:

- Capítulo 1: Introducción.
- Capítulo 2: Estado del arte.
- Capítulo 3: Diseño del trabajo
- Capítulo 4: Desarrollo del trabajo
- Capítulo 5: Integración, pruebas y resultados
- Capítulo 6: Conclusiones y trabajo futuro

## 2 Estado del arte

---

Como se ha mencionado al comienzo de este proyecto, el objetivo de diseño del mismo es la programación de una aplicación que muestre el mayor número de estadísticas posible, siendo su finalidad más importante, maximizar la probabilidad de obtener beneficios de los usuarios en el mundo de las apuestas.

Por ello el primer paso ha sido realizar un estudio para conocer las diferentes aplicaciones que pueden ser similares y estén disponibles para su descarga.

### 2.1 Aplicaciones similares

En este caso, la realidad es que debido a que la práctica de deporte en la actualidad está de moda, existe una gran cantidad de aplicaciones deportivas.

Nos centraremos en las aplicaciones similares a la que vamos a desarrollar, las cuales ofrezcan resultados y algún tipo de estadísticas deportivas, mostrando cuales son sus puntos fuertes y debilidades.

#### 2.1.1 SofaScore



**Figura 1-1: Logo SofaScore**

Ésta es una aplicación disponible en Play Store y cuenta con más de cinco millones de descargas. Se trata de una aplicación que ofrece resultados en directo de 17 deportes distintos de todo el mundo. Los aspectos más remarcables de esta aplicación son:

- + Resultados en directo con avisos
- + Gran variedad de deportes
- + Optimizado para smartwatch

- Demasiada publicidad
- Estadística pobre
- Errores en notificaciones de gol

### 2.1.2 Predicción de fútbol



**Figura 1-2: Logo Predicción de fútbol**

Aplicación que cuenta con más de medio millón de descargas de la plataforma Play Store. En este caso, la aplicación se centra únicamente en el fútbol, ofreciendo predicciones para los partidos disponibles. Los aspectos más remarcables de esta aplicación son:

- + Ofrece muchas ligas
- + Predicciones basadas en estadísticas
- Demasiada publicidad
- Estadística no disponible, únicamente ofrece porcentaje de victorias de cada equipo
- Interfaz muy pobre

### 2.1.3 Predicción de fútbol Forebet



**Figura 1-3: Logo Predicción de fútbol Forebet**

Aplicación de predicciones de fútbol con más de cien mil descargas en el Play Store. Se trata de predicciones matemáticas para la mayoría de tipos de apuestas con un estilo muy parecido a la anterior. Los aspectos más remarcables de esta aplicación son:

- + Muchos mercados (casas de apuestas) para cada partido
- + Ofrece el coeficiente de beneficios más alto de las casas de apuestas
- Interfaz pobre

- No ofrece estadísticas servibles
- Ofrece pocos días para ver los partidos

En general existe un gran número de aplicaciones parecidas, cuyas características son muy similares, aunque en su gran mayoría se tratan de aplicaciones bastante pobres, que parecen ser antiguas o que no están siendo actualizadas, pues la calidad del diseño deja mucho que desear.

Por lo tanto, una vez realizado el estudio de la recopilación de las aplicaciones más populares dentro de la búsqueda “estadísticas deportivas”, se puede concluir que se trata de un sector con gran margen de mejora para crear una aplicación nueva, más interesante y más competitiva, que ofrezca al usuario no solo porcentajes sino también datos en bruto, medias totales y estadísticas más a corto plazo.

Creando una interfaz más visible y fácil de usar, también mejorará mucho el aspecto de la aplicación. Estos son algunos de los puntos que ofrece el presente proyecto, los cuales se desarrollaran más adelante.



## 3 Diseño

---

### 3.1 Objetivos de diseño

El objetivo de este trabajo es el diseño de una aplicación para el sistema operativo Android, mediante la cual se muestren todo tipo de estadísticas deportivas de fútbol y baloncesto, fundamentalmente estudiadas para su uso en el mundo de las apuestas, aunque pueden ser útiles para muchos fines.

Además, en este proyecto una gran parte de trabajo se basa en el tratamiento masivo de datos puesto que la base de datos utilizada y creada desde cero cuenta con más de cinco mil filas de datos y con una cantidad variable de columnas que en ningún caso son menos de diez.

Por lo tanto, el trabajo realizado no ha consistido únicamente en el diseño de la aplicación Android, sino que también se ha realizado una importante tarea para crear el entorno apropiado para el correcto funcionamiento de la citada aplicación.

Tras analizar con anterioridad la posible competencia, se consideró que se puede desarrollar una aplicación que mejore la mayoría de las aplicaciones existentes en la actualidad, con posibilidades de ser de las primeras en Google Play Store. Con las mejoras antes mencionadas y otras que se desarrollarán más adelante, se buscó crear una aplicación muy sencilla que sea capaz de conseguir que un usuario en cuestión de pocos segundos haya visualizado de forma general un evento deportivo y se haya podido hacer una idea estadística de lo que puede suceder.

### 3.2 Requisitos de diseño

En este apartado se exponen los diferentes requisitos que tendrán que cumplir los dispositivos móviles y tabletas para que la aplicación desarrollada en este proyecto pueda funcionar correctamente.

### **3.2.1 Sistemas operativos compatibles**

Antes de comenzar a desarrollar la aplicación para cualquier tipo de dispositivo móvil, se debe considerar cuáles son los sistemas operativos para los que estará disponible. Este es un punto clave puesto que, dependiendo de esta decisión, se tendrá que diseñar y desarrollar la aplicación de una forma u otra, ya que la elección del sistema operativo condiciona tanto el diseño como los aspectos técnicos.

En este caso en particular se eligió desarrollar la aplicación en el sistema operativo Android.

Una de las razones más importantes por las que se ha decidido escoger Android, es porque es un sistema operativo libre y de código abierto, lo que significa que se pueden obtener todas las herramientas para el diseño de la aplicación de forma gratuita desde el portal oficial de desarrolladores. Además, son herramientas multiplataforma, lo que significa que se puede trabajar tanto en Windows, como en Mac OS o Linux, gran ventaja que no ofrecen por ejemplo, las herramientas para crear una aplicación para Apple, puesto que son de pago y únicamente se podrá trabajar en dispositivos con su sistema operativo. Otro gran punto a favor, es la posibilidad de testear la aplicación en dispositivos sin necesidad de tener licencia de desarrollador, cosa que es imposible para una aplicación en Apple. Otro punto importante que apoya la decisión, es que una vez acabada la aplicación y a la hora de subirla a su correspondiente “App Store”, el coste de la licencia de desarrollador en Android es de un pago único de 25\$ y en el caso de Apple es de 99\$, con lo que la diferencia de precio es bastante notoria.

Estos son algunos de los motivos por los cuales Android cuenta con un mayor número de desarrolladores de aplicaciones. Además, esto ayuda a tener un gran ecosistema del sistema operativo, lo que es de gran ayuda cuando se tiene un problema ya que se podrá acudir a la comunidad, especialmente los foros, para ser resuelto en poco tiempo, o, como ocurre en la mayor parte de los casos, para encontrar la solución de forma inmediata debido a que alguien ya ha planteado el mismo problema con anterioridad.

Aunque la aplicación inicialmente está pensada únicamente para usuarios de España, en un futuro se podría extender mundialmente, por ello se están realizando estudios para ver la



distribución de dispositivos, que existen dependiendo del sistema operativo con el que operan.

Smartphone OS Sales Share (%)							
Germany	3 m/e Jan'16	3 m/e Jan'17	% pt. Change	USA	3 m/e Jan'16	3 m/e Jan'17	% pt. Change
Android	74.2	75.5	1.3	Android	58.2	56.4	-1.8
iOS	19.3	21.3	2.0	iOS	39.1	42	2.9
Windows	5.9	2.9	-3.0	Windows	2.6	1.3	-1.3
Other	0.7	0.2	-0.5	Other	0.1	0.3	0.2
GB	3 m/e Jan'16	3 m/e Jan'17	% pt. Change	China	3 m/e Jan'16	3 m/e Jan'17	% pt. Change
Android	52.6	54.4	1.8	Android	73.9	83.2	9.3
iOS	38.6	43.3	4.7	iOS	25.0	16.6	-8.4
Windows	8.6	1.9	-6.7	Windows	0.9	0.1	-0.8
Other	0.2	0.3	0.1	Other	0.3	0.1	-0.2
France	3 m/e Jan'16	3 m/e Jan'17	% pt. Change	Australia	3 m/e Jan'16	3 m/e Jan'17	% pt. Change
Android	71.9	72.9	1.0	Android	52.6	55.7	3.1
iOS	19.3	24.2	4.9	iOS	41.2	42.4	1.2
Windows	7.8	2.8	-5.0	Windows	5.4	1	-4.4
Other	0.9	0.2	-0.7	Other	0.8	0.8	0.0
Italy	3 m/e Jan'16	3 m/e Jan'17	% pt. Change	Japan	3 m/e Jan'16	3 m/e Jan'17	% pt. Change
Android	78.1	79	0.9	Android	48.7	49	0.3
iOS	14.4	15.8	1.4	iOS	50.3	49.5	-0.8
Windows	7.2	4.4	-2.8	Windows	0.5	1.5	1.0
Other	0.3	0.8	0.5	Other	0.5	0	-0.5
Spain	3 m/e Jan'16	3 m/e Jan'17	% pt. Change	EU5	3 m/e Jan'16	3 m/e Jan'17	% pt. Change
Android	87.8	89.4	1.6	Android	72.9	74.3	1.4
iOS	11.4	10.2	-1.2	iOS	20.3	22.7	2.4
Windows	0.8	0.4	-0.4	Windows	6.4	2.7	-3.7
Other	0	0	0.0	Other	0.5	0.3	-0.2

**Figura 2-1: Distribución de dispositivos por SO en 2017**

Como se puede observar, España es el caso más extremo si hablamos de diferencia de uso entre Android e iOS. Además, se puede decir que mundialmente estos son los dos sistemas operativos predominantes en lo que a dispositivos móviles se refiere. Finalmente, como como visión general se puede observar que en ninguno de los países escogidos para este estudio predomina el uso del sistema operativo iOS frente a Android, a excepción de Japón, donde podríamos decir que ambos sistemas operativos están parejos.

La razón de esta distribución reside en la competencia que presenta un sistema operativo frente a otro, pero la diferencia está en que Android cuenta con una gran cantidad de compañías que diseñan dispositivos compatibles con este OS, al contrario que iOS puesto que aquí únicamente Apple hace uso de este sistema operativo.

### 3.2.2 Versiones compatibles

Después de haber dejado claro cuál es la elección en cuanto a qué sistema operativo usar, seguidamente se decidió a partir de qué versión de este sistema operativo estará disponible nuestra aplicación.

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.8%
4.1.x	Jelly Bean	16	3.2%
4.2.x		17	4.6%
4.3		18	1.3%
4.4	KitKat	19	18.8%
5.0	Lollipop	21	8.7%
5.1		22	23.3%
6.0	Marshmallow	23	31.2%
7.0	Nougat	24	6.6%
7.1		25	0.5%

**Figura 2-2: Distribución móviles Android por versión de SO en Marzo de 2017**

La idea es llegar a la mayor cantidad de usuarios posibles, con lo que, a la hora de comenzar con la ejecución del diseño de la aplicación, lo primero y fundamental fue elegir los requisitos mínimos que tenía que soportar el dispositivo para poder instalar la aplicación, en este caso, para maximizar los posibles usuarios, se tomó la decisión de que la aplicación esté disponible para todos los dispositivos Android, a partir de la versión con API 15 o superior.

Esta decisión queda expuesta dentro del proyecto, en el archivo Manifest.xml, donde se podrá modificar este campo y otros.

### 3.2.3 Pantallas compatibles

En primer lugar, antes de pasar a explicar los tipos de pantalla, cabe destacar que se ha decidido realizar esta aplicación para que sea compatible únicamente para Smartphones y tabletas, puesto que se entiende que para otros dispositivos que corren este sistema operativo no es realmente útil.

Tras lo expuesto, debemos saber que Android se ejecuta en diferentes pantallas con diferentes tamaños y características, tal como la densidad de píxeles.

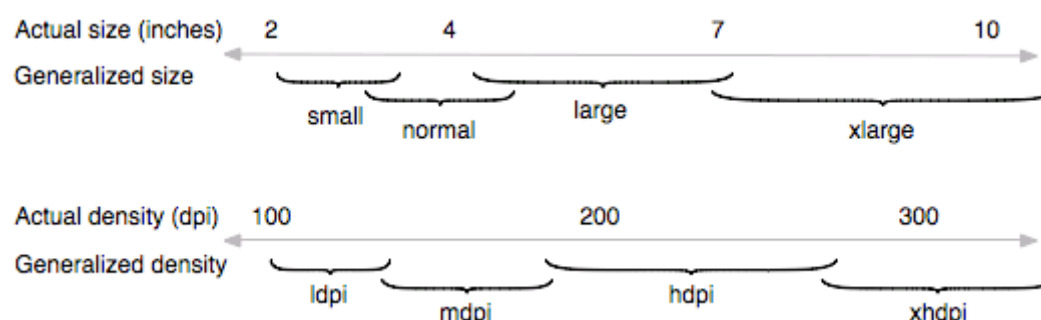
El propio sistema operativo realiza cambios para ajustar la imagen a cada pantalla, pero con un poco de estudio para saber ante qué tipo de pantalla estamos y su elección, se podrá mejorar notablemente la experiencia del usuario, puesto que no solo se busca que la aplicación sea funcional, sino que también cuente con una interfaz llamativa y atractiva al sentido de la vista.

Fundamentalmente los criterios por los que se puede diferenciar una pantalla son el tamaño y la densidad.

Por tamaño se entiende la medida física de la pantalla, ésta se mide en pulgadas en la diagonal de la pantalla. Según este criterio, las pantallas de los dispositivos se pueden clasificar en cuatro tamaños: small, normal, large y extralarge.

La densidad de pantalla es la cantidad de píxeles que hay dentro del área física de la pantalla del dispositivo. Normalmente, ésta se mide en dpi (puntos por pulgada, por sus siglas en inglés). En este caso, nuevamente se puede clasificar en cuatro tipos: ldpi, mdpi, hdpi y xhdi.

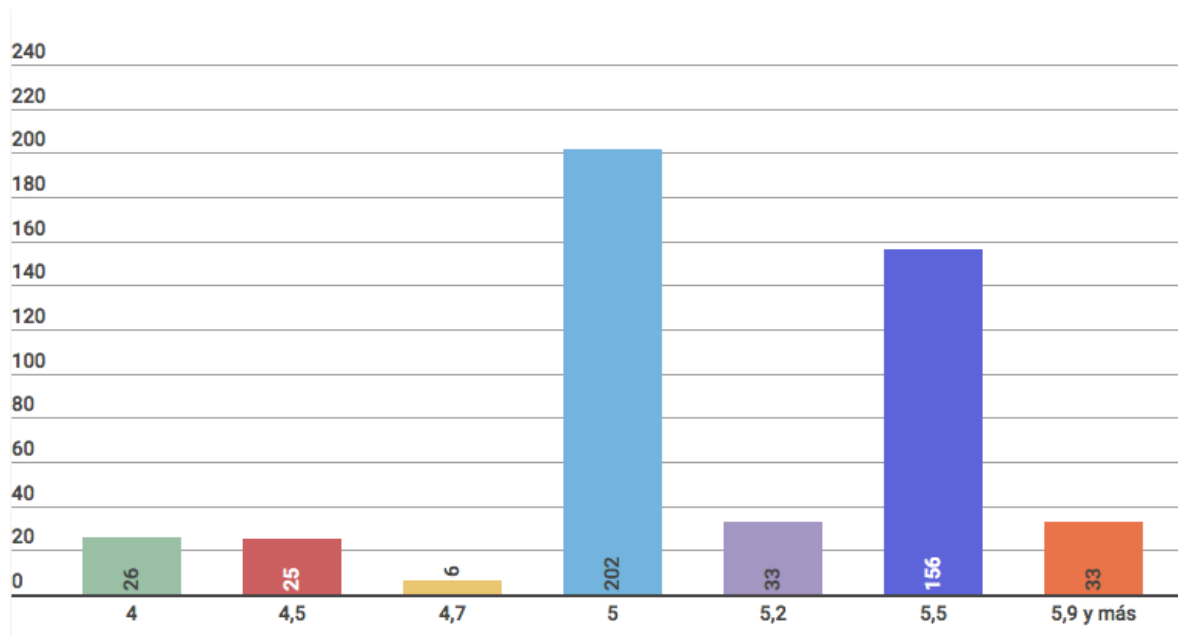
A modo de tener unos valores un poco más generalizados, Android tiene una gráfica en la cual se muestran los rangos en los cual se encasilla cada tipo de pantalla, dependiendo tanto del tamaño como de la densidad de la misma, puesto que existen infinitas combinaciones de tamaño y densidad de pantalla.



**Figura 2-3: Rangos de asignación de tamaños y densidades en función de tamaños y densidades reales**

Una vez conocido cómo se dividen los dispositivos en función de su pantalla, se procederá a decidir para qué pantallas se ajustará nuestra aplicación.

Va a estudiarse cuál es la distribución de dispositivos Android con diferentes pantallas.



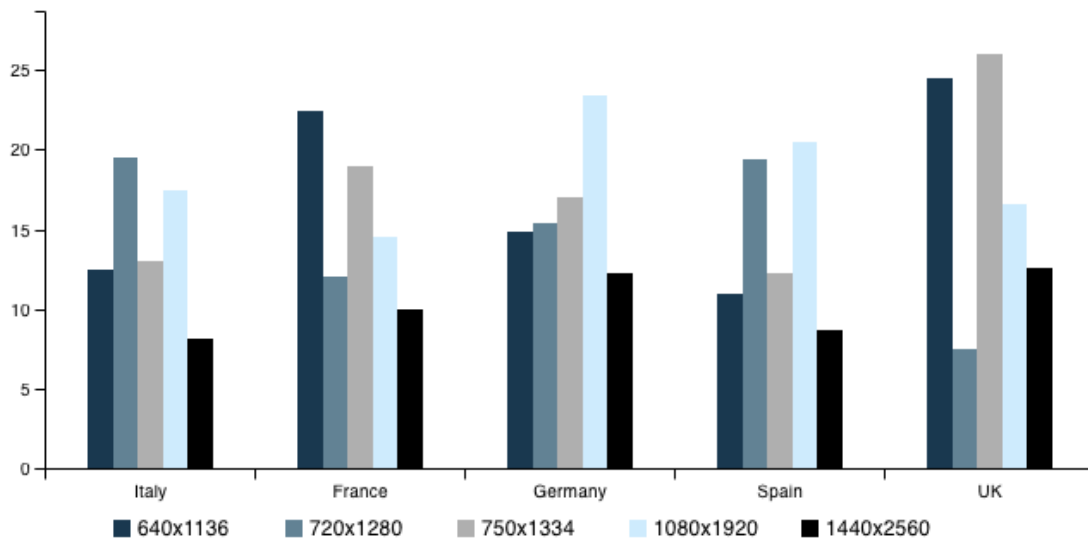
**Figura 2-4: Distribución de los modelos de Smartphone sacados en 2016 por el tamaño de la pantalla**

Como es sabido, en estos tiempos está evolucionando el uso del Smartphone y en lo que al tamaño respecta, ha ido creciendo hasta el punto de que en la actualidad la pantalla de cinco pulgadas es la más usada, como se puede observar en la imagen anterior. La gráfica muestra los modelos de móviles sacados al mercado en el año 2016, segmentados por el tamaño de la pantalla.

Por este motivo, en cuanto a tamaño se ha decidido diseñar la aplicación para los dos tamaños de pantalla de uso generalizado, con el fin de que pueda ser utilizada por la mayoría de usuarios, siendo dichos tamaños el large y el normal.

Además, con estos dos tamaños cumplimos el criterio mostrado anteriormente acerca de la compatibilidad con Smartphones y tabletas, puesto que los otros tamaños suelen estar dirigidos a otros dispositivos que en nuestro caso no serán útiles.

Una vez decidido el tamaño, se procede al estudio de la distribución de los dispositivos en relación a la densidad de pantalla. Esta característica básicamente se define por un número que cuanto mayor sea, mejor y con más nitidez veremos las representaciones en la pantalla de nuestro dispositivo.



**Figura 2-5: Distribución de los Smartphones en 2016 por densidad de pantalla**

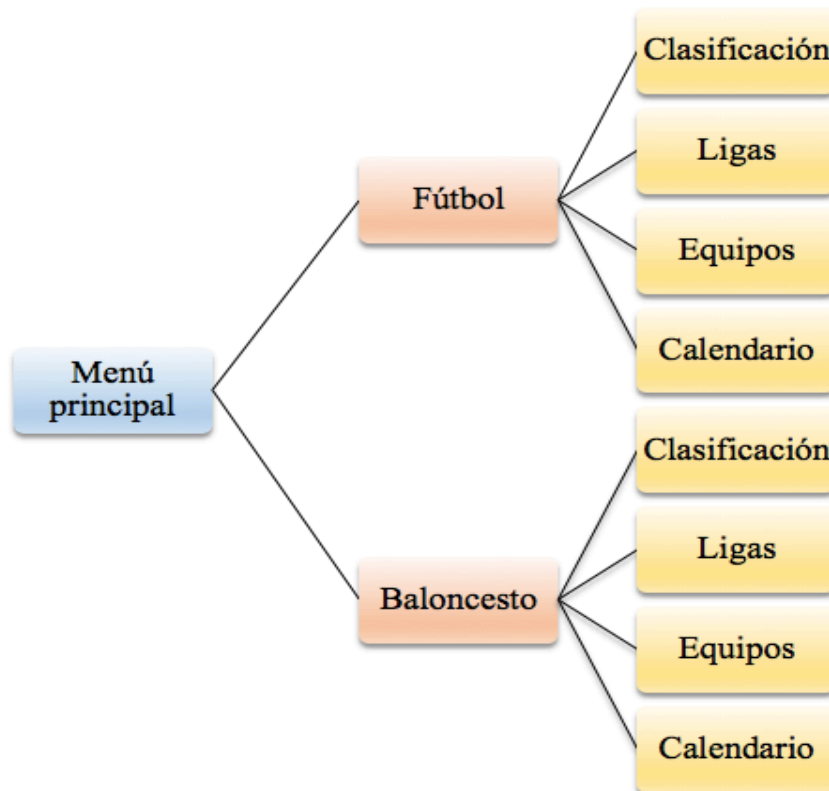
Como se puede observar, en España, la resolución más utilizada es 1080x1920 o también llamada FullHD , seguidas por las pantallas HD. Este punto de estudio es una parte más informativa, puesto que en este caso la aplicación funcionará en todas las densidades de pantalla.

### 3.3 Diagrama de bloques del diseño

Como se ha ido comentando a lo largo del proyecto, se ha tratado de diseñar una aplicación lo más simple e intuitiva posible, con la finalidad de darle al usuario que pretende realizar una apuesta, una información lo más precisa, concisa y útil de la forma más simple posible, para que éste minimice probabilísticamente sus pérdidas.

La estructura que se va a seguir para el diseño de la aplicación, va a ser una estructura en forma de árbol. Mediante esta estructura lo que conseguimos es poder llegar desde el primer menú de elección de deporte a todas las actividades disponibles en la aplicación.

En el siguiente diagrama queda reflejado lo que sería la estructura definida en esta aplicación:



**Figura 2-6: Diagrama de la aplicación**

### **3.4 Estructura de la aplicación**

En este apartado se muestra de forma resumida todos los elementos que componen la aplicación, tanto la denominación de las diferentes pantallas o actividades de la aplicación, como las tablas de la base de datos utilizadas.

### 3.4.1 Servidor remoto

En este servidor contratado para que nos preste sus servicios es donde están almacenados tanto todos los datos en la base de datos, a la cual se accede mediante PhpMyAdmin, como todos los archivos .php creados uno a uno para el acceso a las diferentes tablas de la base de datos y que realizan cada uno diferentes tareas.

### 3.4.2 Splash Screen

Esta es la primera pantalla que aparece al ejecutar la aplicación, la cual permanece durante tres segundos y tiene la función de ser una pantalla de bienvenida a la vez que en el tiempo durante en el que está activa, permite cargarse correctamente a la actividad siguiente.

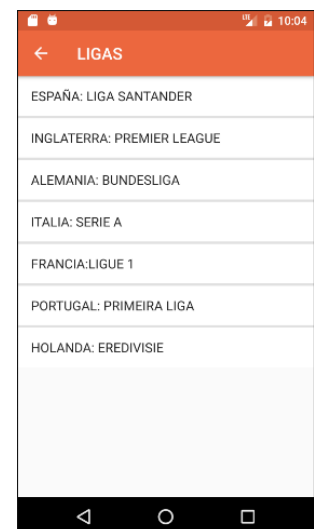
### 3.4.3 Actividad principal

Esta es la primera actividad que tenemos, en ésta decidiremos cual de los dos deportes de la aplicación queremos visualizar. El usuario únicamente tendrá que elegir una opción u otra para pasar a la siguiente actividad.

### 3.4.4 Elección de competición

En el siguiente paso después de haber elegido la opción de deporte, el usuario decidirá cuál de las diferentes ligas disponibles quiere visualizar.

Además, el primer menú de elección de liga nos ayuda a hacer un primer filtrado de información, ayudándonos a la hora de obtener los partidos del día, las clasificaciones y los equipos.



**Figura 2-7: Menú de elección de competición**

### 3.4.5 Menú de competición

Una vez seleccionada una de las opciones anteriores, se accede a esta actividad, en la cual se ofrecen cuatro diferentes opciones para visualizar:

- Clasificación de la liga seleccionada.
- Calendario en el cual se muestran los partidos y resultados con hasta siete días de diferencia hacia atrás y futuros.
- Modificar la liga elegida.
- Mostrar todos los equipos de la liga ordenados alfabéticamente para encontrar rápidamente a un equipo en concreto.



**Figura 2-8: Menú de competición**



## 4 Desarrollo

---

En este apartado, se explican todos los conceptos utilizados para el desarrollo de nuestra aplicación Android. Esta parte tiene un carácter más técnico, ahondando en cómo se programaron los distintos módulos y comentando algunos de los problemas que surgieron durante el desarrollo de la aplicación.

### 4.1 Pasos previos

En primer lugar, antes de comenzar el trabajo y con objeto de abordar con garantías de éxito cada una de las tareas, fue preciso adquirir conocimientos suficientes del desarrollo de Android y familiarizarse con el uso de sus herramientas.

En este primer aprendizaje se llevaron a cabo las siguientes acciones:

- Estudio adquisición de destrezas de varios textos acerca del entorno Android y Java
- Realización de tutoriales, para la familiarización con el entorno de programación
- Estudio y lectura de libros sobre bases de datos
- Creación de bases de datos de prueba con tablas, para instrucción y adiestramiento para su trabajo con ellas.

### 4.2 Herramientas utilizadas

Para el desarrollo de esta aplicación se ha utilizado el entorno de programación Android Studio, el cual es totalmente gratuito. Además, las simulaciones se han realizado en el emulador que trae incluido Android Studio. También ha sido necesario un Smartphone Android y un terminal informático con capacidad suficiente para permitir el desarrollo de los trabajos con suficiente fluidez.

Para la parte de la base de datos se ha utilizado un Hosting privado, en el cual se almacena nuestra base de datos que se ha desarrollado mediante PhpMyAdmin. Además, en este

servidor también se han guardado todos los archivos creados en lenguaje PHP, para la interacción entre la aplicación y la base de datos.

### **4.3 Elementos principales de una aplicación**

A continuación, se explicarán algunos de los aspectos mas básicos que tiene cualquier aplicación Android y tras esto se estudiará la aplicación desde un punto de vista más técnico.

#### **4.3.1 Activity**

Una actividad es cada una de las ventanas que contienen la interfaz de usuario y de la aplicación y en la cual los usuarios pueden interactuar. Las actividades comienzan a funcionar desde el momento que aparecen en la pantalla del usuario. Desde este momento, hasta que termina, la actividad pasa por diferentes etapas, que se conoce como ciclo de vida de una actividad.

Toda actividad cuando comienza, ejecuta el método “onCreate”, dejando la actividad anterior en una pila, por si es necesario volver a hacer uso de ella en el futuro.

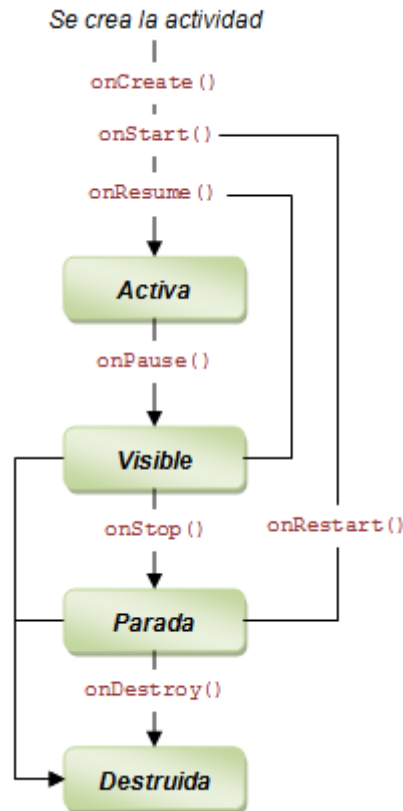
Se debe remarcar que el ciclo de vida de una actividad en Android, difiere bastante al de una actividad en otro sistema, por ejemplo, Windows, puesto que en Android este ciclo de vida lo maneja el propio sistema y no el usuario como ocurre en Windows.

Una actividad tiene cuatro estados fundamentales:

- Activa: La actividad está arriba de la pila, lo que significa que esta visible
- Visible: La actividad está activa pero no tiene el foco, esto ocurre cuando hay otra actividad visible que no ocupa toda la pantalla o es transparente, en el caso de ser tapada completamente pasará al siguiente estado
- Parada: La actividad no está visible. Aún así se deben de guardar los estados de la actividad por si vuelve a esta activa

- Destruída: Cuando la actividad termina invocándose al método “onDestroy()” o el propio sistema la mata.

Aquí se muestra esquemáticamente el ciclo de vida de una actividad:



**Figura 3-1: Ciclo de vida de una actividad**

Una vez visto el ciclo de vida de una actividad, hay que decir que una actividad consta de dos partes, una lógica y una gráfica.

La parte lógica es desarrollada en java y es en la que están todos los métodos que se han explicado anteriormente. La parte gráfica es un archivo XML en el que se declararán todos los elementos que se muestran en pantalla.

Estas dos se unen mediante la función “setContentView()” declarada en la parte lógica.

### 4.3.2 Views

Una vista es una clase que representa una estructura rectangular dentro de la pantalla y tiene las características de la pantalla, permitiendo establecer los Layout, para mostrar objetos o imágenes.

### 4.3.3 Manifest

Este es un archivo XML el cual es necesario que este en el directorio raíz de cualquier aplicación que se desarrolle en Android. Este archivo es el que decide las propiedades que tendrá la aplicación, como por ejemplo en nuestro caso, tendremos que darle permisos para que la aplicación pueda acceder a internet. Y esto se logra añadiendo etiquetas de “uses-permission”:

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

## 4.4 Elementos principales de las bases de datos

En este apartado se explica cómo crear una base de datos y sus partes:

### 4.4.1 Base de datos

Una vez que se disponga de un servidor, se procederá a crear una base de datos. En este caso se creó mediante el asistente que incluye nuestro hosting MySQL. Los pasos para ello incluyen abrir el asistente y rellenar el nombre que queremos asignar a la base de datos, y opcionalmente para poder acceder remotamente a la base de datos, también sería necesario crear un usuario que tenga privilegios para poder acceder a ella.

### 4.4.2 Tablas

Una vez creada la base de datos, se accede a ella mediante phpMyAdmin y lo primero que debe hacerse para comenzar a introducir información es crear una tabla. Todo esto se puede hacer de dos formas, bien escribiendo directamente en la “consola” de SQL que tiene phpMyAdmin o bien mediante los asistentes que nos facilitan mucho las cosas. Hay que definir el nombre de la tabla y cada una de las columnas que queremos que tenga la tabla. Para cada columna hay que rellenar información, tal como qué tipo de dato va a ser, la longitud máxima o si puede ser nulo, además de otros. Es importante decir que siempre tiene que haber una columna que haga de clave primaria, lo que hará que no puedan duplicarse registros.

## 4.5 Menús de la aplicación

La aplicación consta de tres menús principales: elección de deporte, elección de competición, opciones de visualización. Además, se puede añadir un cuarto menú, el menú de elección de fechas que se encuentra dentro de la actividad del calendario.

Para esta primera versión no se ha prestado excesiva atención a la parte más visual, sino que se ha pretendido conseguir una aplicación funcional, fácil e intuitiva para el perfil de usuario definido anteriormente.

En el primer menú se ha optado por poner dos ImageButton, cada uno con una pelota distintiva de su deporte.

En el menú de elección de competición se muestra como una ListView las diferentes competiciones que están disponibles, en este caso este menú se crea dependiendo de los objetos que haya en una tabla de la base de datos.

El menú de elección de visualización ha sido creado mediante simples botones en los cuales se ha puesto el nombre de cada apartado al que se va a redireccionar al ser pulsado, para ofrecer la mayor simplicidad.

Por último, el menú de elección de fechas es un menú de opciones en el cual el usuario tiene la posibilidad de elegir la fecha en un calendario entre los siete días futuros y pasados para obtener los partidos de ese día.

## **4.6 Splash Screen**

Se trata de la primera imagen que aparece cuando se ejecuta la aplicación y tiene dos misiones, la primera es para dar la bienvenida al usuario antes de mostrar la aplicación en sí. La otra misión es obtener un tiempo que utiliza la aplicación para cargar las funciones que va a utilizar posteriormente y de esta manera hacer ágil la aplicación sin demoras en las ejecuciones.

En general, se suele colocar una imagen con el logo de la aplicación como Splash Screen. En este caso al no tener un logo definido se ha optado por poner una imagen de un color azul plano, que nos permite hacer la precarga de las funciones.

Para colocar la Splash Screen se han seguido los siguientes pasos:

- Crear una nueva actividad llamada Splash Screen
- Asignar la forma y colores que queremos en el RelativeLayout de nuestro XML de la actividad
- Crear un temporizador en la parte Java, que en nuestro caso son tres segundos, para que muestre nuestro Splash Screen
- En el AndroidManifest decir a la aplicación que esta sea la primera actividad que se ejecute una vez la aplicación empieza a correr

## **4.7 Desarrollo base de datos**

En este caso se ha utilizado una base de datos SQL almacenada en un hosting privado a la cual se accede mediante phpMyAdmin. Para la creación de la citada base de datos se ha hecho un estudio lo más detallado posible de todos los datos que se pueden recoger de un partido. Por ejemplo, la tabla donde se almacenan los datos de cada partido, la componen un total de 49 columnas de información que posteriormente también son usadas para actualizar otras tablas como las tablas de los equipos individuales. Todo este trabajo se ha

llevado a cabo mediante consultas SQL y scripts PHP creados para optimizar las tareas. Para conectar de manera remota con la base de datos, se han utilizado las herramientas de las librerías PHP en este caso específicamente SQLi. Posteriormente, se envían los datos obtenidos por las consultas a la aplicación codificados por JSON.

## 4.8 Desarrollo de menú de elección de ligas

Este es uno de los primeros menús que nos ofrece la aplicación para poder obtener la información más precisa para el usuario.

En este caso las ligas que aparezcan en este menú serán aquellas ligas que actualmente están en la base de datos. Para mostrar este menú se han obtenido todas las ligas con las que contamos en la actualidad y en la actividad de la aplicación se ha creado una cadena en la cual se almacenan todas ellas. Además de crear la cadena, en la función `onCreate()` se ha añadido la función `setOnClickListener()`, función que escucha durante su ejecución para extraer información acerca de dónde ha pulsado el usuario, para realizar una acción, que en este caso, pulse la opción que pulse será redireccionado a la misma siguiente actividad, la diferencia es la información que se le pasa a la siguiente actividad mediante el método `putExtra()`, consiguiendo que en las actividades posteriores se muestre la información de la liga seleccionada.



En la parte de diseño del Layout se crea un simple `ListView`, con el que se mostrará toda la información de la cadena declarada en la parte java, con un formato ya definido en las librerías de Android (`simple_list_item_1`) que hacen que nuestra actividad tenga el aspecto que resulta.

```
ListAdapter adapter = new ArrayAdapter<String>(Ligas.this,
android.R.layout.simple_list_item_1, ligas);
```

```
lista.setAdapter(adapter);
```

Donde la variable “lista” se ha declarado anteriormente como ListView en el archivo Java y posteriormente se ha relacionado mediante la función findViewById() con el módulo ListView declarado en el Layout del archivo XML.

## 4.9 Botón retroceder

Se ha creado un botón que está presente en todas las actividades mediante el cual se puede ir a la actividad anterior. Este botón se ha creado para la comodidad de los usuarios, y para darle la sensación de que siempre puede interactuar con la aplicación, evitando de esta forma que en ningún momento se esté en una actividad en la cual no pueda interactuar con ella.

Para realizar lo expuesto tenemos funciones que nos ayudan a lograr este fin. En el archivo java lo primero que hacemos es comprobar si la barra de acciones esta activada y en ese caso, el siguiente paso es activar la imagen de la flecha hacia atrás.

```
if(getSupportActionBar() != null){  
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);  
}
```

Una vez hecho esto, lo que necesitamos ahora es decir qué queremos que haga la aplicación cuando la flecha sea presionada. Esto se realiza en la función onOptionsItemSelected(), mediante la cual, con un switch tenemos todas las acciones que se pueden realizar en una actividad, de modo que tras poner un case apuntando al ítem, simplemente se llama a la función onBackPressed(), la que llevará al usuario a la actividad anterior.

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case android.R.id.home:  
            onBackPressed();  
            return true;  
        default:  
            return super.onOptionsItemSelected(item);  
    }  
}
```

## 4.10 Desarrollo Clasificación

Para realizar esta actividad se ha creado un archivo PHP expresamente para esta acción. La actividad tiene que pasar a nuestro archivo PHP alojado en el hosting, datos adjuntos para que realizando comparaciones, se obtenga la competición. Con los datos de competición



obtenidos, posteriormente se conecta y realiza una consulta mediante SQL a la base de datos, obteniendo todos los demás datos necesarios que más tarde se mandarán para poder ser mostrados en la aplicación.

Una vez realizada la consulta, los datos resultantes se guardan en un array y se codifican para ser enviados mediante JSON, haciendo uso de la función `json_encode`.

Al recibir esta información, la aplicación tiene que parsear los datos para poder utilizarlos correctamente. Esta acción se realiza mediante un archivo llamado `JSONParser.java` creado para este fin y al cual llamarán todas las actividades que beban de la base de datos, puesto que el retorno de los datos será siempre en formato JSON.

En nuestro caso, actualmente únicamente se realizan consultas mediante el método GET, puesto que únicamente interesa hacer consultas a la base de datos sin modificar ningún cambio. Debido a que en futuras actualizaciones se ha pensado en añadir un formulario para registro y login entre otras mejoras, también se han añadido los pasos para consultas por el método POST.

En este archivo lo que hacemos es una consulta, la cual se ha creado para que si queremos añadir parámetros, se añadan automáticamente a la URL de forma correcta, añadiendo tras la URL una `”?”` en caso de haber parámetros y si hay más de uno, añade `”&”` entre parámetro y parámetro para formar correctamente la URL completa y los parámetros lleguen a la consulta.

Una vez parseados los datos correctamente, se pasa a la parte de diseño. Todos los datos serán mostrados en el archivo XML de la actividad mediante un `ListView`, pero para que se muestren de la forma adecuada, se ha creado otro archivo XML con el diseño deseado mediante `TextViews`. Haciendo uso de la función `simpleAdapter` que incluyen las librerías de Android, se colocan los datos parseados anteriormente en el formato del archivo de diseño que hemos escrito y finalmente este formato es pasado al `ListView` de la actividad, mostrando los datos por filas exactamente como queremos.

```

ListAdapter adapter = new SimpleAdapter(
    Clasificacion.this,
    datosParseados,
    R.layout.diseño_a_mostrar,
    new String[]{
        TAG_ID,
        TAG_POSICION,
        TAG_NOMBRE,
        TAG_PARTIDOS,
        TAG_GOLES_A_FAVOR,
        TAG_GOLES_EN_CONTRA,
        TAG_PUNTOS,
    },
    new int[]{
        R.id.single_post_tv_id,
        R.id.single_post_tv_pos,
        R.id.single_post_tv_nombre,
        R.id.single_post_tv_partidos,
        R.id.single_post_tv_goles_a_favor,
        R.id.single_post_tv_goles_en_contra,
        R.id.single_post_tv_puntos,
    });

lista.setAdapter(adapter);

```

Además, cuando el usuario pulsa en cualquier equipo le lleva a su perfil mostrando sus estadísticas más detalladas. Esto se realiza mediante la función `setOnClickListener()`, que devuelve la posición del elemento que ha sido pulsada y con esta información y realizando comparaciones, permite conocer qué equipo se ha seleccionado y hay que mostrar en la siguiente actividad.

## 4.11 Desarrollo actividad calendario

Este apartado se divide en dos partes que son las más importantes en esta actividad, la parte de diseño para mostrar correctamente los partidos y la parte de elección de fecha en el calendario.

### 4.11.1 Mostrar partidos

Este apartado fue el que más problemas dio durante su desarrollo, puesto que se buscaba una estructura que Android no tenía incluida en sus librerías por lo que fue necesaria desarrollarla desde cero, para obtener el resultado que se muestra en el siguiente esquema:

- Partido futuro:

Nombre equipo 1		Hora encuentro
Nombre equipo 2		

- Partido terminado:

Nombre equipo 1	Goles	Finalizado
Nombre equipo 2	Goles	

Las dificultades en cuanto al diseño eran varias, puesto que hay que hacer una consulta para que la base de datos devuelva los datos necesarios y posteriormente crear el Layout para mostrar los datos de esta forma.

El mayor problema encontrado fue el crear las dos primeras columnas con dos filas y la tercera con una única fila. Tras muchas pruebas y distintas ideas, se llegó a la solución integrando en el archivo XML del simpleAdapter, diferentes RelativeLayout que lo que nos permite esta etiqueta es integrar diferentes espacios dentro de una misma actividad, logrando exactamente el diseño que pretendíamos desde el principio, como podemos observar:



**Figura 3-2: Ejemplo de uso RelativeLayout**

Por otro lado, para modificar los campos dependiendo de si el partido ya ha sido jugado o no, se hacen varias consultas a la base de datos, utilizando los parámetros que se le pasan por la URL y mediante condiciones, la consulta nos devuelve unos datos u otros.

```

105 while($row = mysqli_fetch_array($form)) {
106     if($row["Competicion"]=="1" || $row["Competicion"]=="2"){
107         $local=mysqli_query($con,"SELECT Nombre FROM `TablaEquipos_1` Where idEquipo='$seguipo' ");
108         $loc= mysqli_fetch_array($local);
109         $visitante=mysqli_query($con,"SELECT Nombre FROM `TablaEquipos_1` Where idEquipo='$vvisitante' ");
110         $vis= mysqli_fetch_array($vvisitante);
111     }
112     else{
113         $local=mysqli_query($con,"SELECT Nombre FROM `TablaEquipos_2` Where idEquipo='$local' ");
114         $loc= mysqli_fetch_array($local);
115         $visitante=mysqli_query($con,"SELECT Nombre FROM `TablaEquipos_2` Where idEquipo='$vvisitante' ");
116         $vis= mysqli_fetch_array($vvisitante);
117     }

```

**Figura 3-3: Código PHP para la consulta de partidos**

### 4.11.2 Calendario Partidos

Una de las utilidades más interesantes que se ofrecen con esta aplicación es la opción de modificar el día que se quiere visualizar en el calendario, en este caso se puede elegir entre siete días atrás y siete días futuros, dando la oportunidad, tanto de ver los resultados anteriores, como los enfrentamientos futuros que habrá.

Sin duda, el desarrollo de esta funcionalidad fue el más complicado dentro de todo el proyecto, pero es una funcionalidad muy necesaria para la consulta de estadísticas, o en su caso para consulta de resultados de los partidos precedentes.

La primera dificultad encontrada, fue el crear un icono del tipo Spinner en el que aparecieran todas las opciones de días que podemos consultar. En este caso, se crearon 15 etiquetas ítem en el archivo XML para que aparecieran dentro del Spinner.

Tras este comienzo, el siguiente problema fue el añadir datos variables de fechas a este icono de opciones. Se buscaba mostrar los siete días anteriores y posteriores a la fecha actual, con lo que, si se entra al día siguiente a la aplicación, estas fechas deberán haber cambiado.

```
> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:background="#f0f0f0"
    android:weightSum="1"
    android:orientation="vertical">
    > <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    >
    > <TextView
        android:id="@+id/single_post_tv_local"
        android:layout_width="103dp"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:paddingBottom="2dip"
        android:padding="10dp"
        android:textColor="#333"
        android:textSize="12dp"
        android:layout_weight="0.30"
        android:textStyle="bold"
        android:text="nombre" />
    > <TextView
        android:id="@+id/single_post_tv_res_local"
        android:layout_width="103dp"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:paddingBottom="2dip"
        android:padding="10dp"
        android:textColor="#333"
        android:textSize="12dp"
        android:layout_alignParentRight="true"
        android:layout_weight="0.30"
        android:textStyle="bold"
        android:text="vis" />
    > </RelativeLayout>
```

Figura 3-4: Código XML de la actividad Calendario

Una vez conseguidos estos dos pasos, se habría concluido la parte de diseño de la aplicación y se entra con la parte de envío de parámetros para consultar a la base de datos.

Por último, decir que la gestión del día actual fue la opción más costosa de todas, puesto que en este día pueden ocurrir dos cosas, que haya un partido futuro o que ya se haya jugado.

Una vez explicado la mayoría de desarrollos que se han realizado para este trabajo, cabe destacar que estos se han tenido que hacer dos veces debido a que en ambos deportes hay que mostrar un conjunto de datos similares, pero difieren entre unos y otros, con lo que no se pueden utilizar las mismas plantillas. Los archivos XML sí que se han podido utilizar, puesto que estos sí que van a tener la misma forma. Estos serán iguales por el simple motivo de que toda la aplicación tiene el mismo aspecto y todo sigue unas líneas comunes.



## 5 Integración, pruebas y resultados

---

### 5.1 Añadir aplicación al Play Store

Una vez finalizada la aplicación y comprobado que se ejecuta correctamente, el siguiente paso es generar el fichero APK para publicarlo en la plataforma de Google Play Store.

#### 5.1.1 Generación de APK

Para generar el archivo APK, hay que abrir el programa Android Studio y en el menú de opciones acceder a la opción de “Build”. Aquí dentro se escoge la opción de “Generate signed APK”.

Este es un punto importante, puesto que para que una aplicación sea aceptada en la plataforma deberá contar con certificado digital. El proceso de firma es de gran utilidad por motivos de seguridad y además aporta un punto de seguridad para el desarrollador, ya que será el único que podrá modificarla.

El siguiente paso es crear un Key Store, aquí deberemos rellenar un conjunto de datos personales, además de una contraseña. Esta servirá para futuras exportaciones de la aplicación con lo que habrá que guardarla en un lugar seguro.

Una vez creado el Key Store se debe exportar el fichero APK al equipo. Ya tenemos listo el fichero para su publicación.

#### 5.1.2 Cuenta de desarrollador

Para añadir una aplicación a la plataforma es necesario contar con una cuenta de desarrollador. Para obtenerla se deben abonar 25€ y rellenar datos personales. Mediante esta cuenta se podrán publicar varias aplicaciones.

Una vez realizados todos los pasos se tiene acceso a las opciones de consola del desarrollador, donde se publicará la aplicación.

## **5.2 Subir la aplicación a la plataforma**

En la cuenta de desarrollador se accede a la consola, donde se deben seguir los siguientes pasos para la publicación:

- Hacer click en “Add New Application”. Se debe introducir el idioma, el nombre de la aplicación y adjuntar el archivo APK.
- Indicar la versión de la aplicación que se va a publicar( test alfa, beta o modo de producción) .
- En la pestaña “Store listing” se deben rellenar los siguientes puntos:
  - Descripción completa.
  - Texto de promoción.
  - Icono.
  - Pantallazos.
  - Categoría.
  - Privacidad.
  - Palabras clave.
- En la pestaña “Pricing & distribution” hay que especificar el precio de la aplicación, que en este caso es gratuita, y los países donde se distribuirá, que en este caso se escoge el máximo posible, 141 países.

Una vez completados los puntos anteriores, ya se puede publicar la aplicación, desactivando la opción de borrador.

## **5.3 Plan de pruebas**

Para comprobar que la aplicación tiene un correcto funcionamiento, se ha efectuado el siguiente plan de pruebas:



<b>Acción</b>	<b>Resultado esperado</b>
Descargar App en Smartphone	Ejecuta correctamente
Descargar App en tableta	Ejecuta correctamente
Pulsar un deporte	Muestra la elección de liga de ese deporte
Pulsar Clasificación	Salen clasificación de la liga seleccionada
Pulsar Botón atrás	Aparece la página anterior
Pulsar Calendario	Muestran los partidos de hoy
Pulsar en el icono del calendario y elegimos otra fecha	Muestran los partidos que hay para la fecha elegida
Entrar a la App con la conexión apagada	Mensaje de error + mensaje de comprueba tu conexión a Internet
Entrar a la app a la parte que calendario dos días distintos	Aparecen los días del calendario correctamente dependiendo de la fecha
Se modifica el día en el calendario	Muestran los partidos del día elegido en el formato correcto



## 6 Conclusiones y trabajo futuro

---

### 6.1 Conclusiones

Una vez finalizado este proyecto, se puede afirmar que se han cumplido todos los objetivos que se marcaron al comienzo del mismo.

El objetivo principal del proyecto era desarrollar una aplicación mediante la cual se mostrasen todo tipo de estadísticas enfocadas al fútbol y baloncesto, haciendo uso de una base de datos creada a mano, donde se almacenan todos los datos de la temporada.

La aplicación tiene las siguientes funcionalidades:

- Clasificaciones de fútbol de las ligas más importantes a nivel europeo
- Clasificaciones de baloncesto de las ligas más importantes a nivel mundial
- Estadísticas detalladas de cada equipo
- Calendario para mostrar los partidos que hay en el día
- Función para filtrar por días el calendario

Desde el punto de vista educativo, este TFG ha servido para adquirir conocimientos suficientes acerca de:

- Bases de datos
- Lenguaje SQL
- Lenguaje PHP
- Lenguaje XML
- Lenguaje Java
- Programación de aplicaciones Android
- Manejo de entorno de programación Android Studio
- Preparación y publicación de aplicaciones en Google Play Store

## 6.2 Trabajo futuro

Durante el desarrollo de este proyecto se han realizado las funcionalidades más importantes y prioritarias de la aplicación, no obstante se han ido pensando nuevas funcionalidades que pueden mejorar nuestra aplicación. Algunas de éstas serían:

- Disposición de marcadores en directo para aumentar las veces que un usuario entre a consultar datos a la aplicación
- Crear un login, para que cada usuario tenga su cuenta
- Añadir imagen con escudo de cada equipo
- Crear un botón mediante el cual los usuarios puedan elegir sus partidos favoritos
- Crear notificaciones push para enviar información a los usuarios de sus partidos favoritos
- Estudiar, analizar y optimizar el consumo de batería y datos
- Adaptar aplicación para iOS
- Poder logearse desde las redes sociales y poder compartir con otros contactos
- Estudio para monetizar la aplicación (versiones freemium, con publicidad,...)
- Aplicación para diferentes idiomas
- Mejoras en diseño para hacer la aplicación más atractiva
- Estudiar una campaña de marketing para promocionar la aplicación

# Referencias

---

- [1] Joan Ribas Lequerica, ANAYA MULTIMEDIA, “DESARROLLO DE APLICACIONES PARA ANDROID”, Edición 2016.
- [2] José Dimas Lujan Castillo, RC LIBROS, ”Android: Aprende desde cero a crear aplicaciones”, 2015
- [3] <https://developer.android.com/>
- [4] <http://aprendeandroid.com/>
- [5] <http://stackoverflow.com/>
- [6] <http://gsmarena.com/>
- [7] <https://elandroidelibre.elespanol.com/>
- [8] <https://deviceatlas.com/>
- [9] <https://www.xataka.com/>
- [10] <https://play.google.com/store>

## **Glosario**

---

<b>API</b>	Application Programming Interface
<b>APK</b>	Android Application Package
<b>Dpi</b>	Dots Per Inch
<b>DSLab</b>	Digital System Laboratory
<b>EPS</b>	Escuela Politécnica Superior
<b>hdpi</b>	High Resolution Image
<b>ldpi</b>	Low Resolution Image
<b>mdpi</b>	Medium Resolution Image
<b>UAM</b>	Universidad Autónoma de Madrid
<b>Xhdpi</b>	Extra High Resolution Image
<b>OS</b>	Operating System